

## Software Design Document

# Oficjalna aplikacja 14. Sesji Linuksowej na system Android

**Julian Jurec**

*Akademickie Stowarzyszenie Informatyczne*

## Spis treści:

1. Wprowadzenie
  1. Definicje
  2. Cel
  3. Zakres
2. Wymagania
3. Architektura Systemu
4. Architektura Aplikacji
5. Funkcjonalności
6. Interfejs użytkownika

# **Wprowadzenie**

## **Definicje**

### **Akademickie Stowarzyszenie Informatyczne**

Akademickie Stowarzyszenie Informatyczne (w skrócie: ASI), to organizacja studencka działająca na Politechnice Wrocławskiej, skupiająca pasjonatów informatyki, elektroniki oraz dziedzin pokrewnych. Nasza działalność opiera się przede wszystkim na organizowaniu wykładów, warsztatów i konferencji.

### **Android**

Android to system operacyjny z jądrem Linux dla urządzeń mobilnych takich jak telefony komórkowe, smartfony, tablety (tablety PC) i netbooki. W 2013 roku był najpopularniejszym systemem mobilnym na świecie.

### **Flask**

Flask jest microframeworkiem do aplikacji webowych języka Python opartym o Werkzeug, Jinja 2 i dobre intencje. Jest w pełni open-source'owym rozwiązaniem na licencji BSD. Umożliwia łatwe i szybkie postawienie API i prostych aplikacji webowych.

### **RESTful API**

Representational State Transfer (ang. zmiana stanu przez reprezentację) - styl architektury oprogramowania wywiedziony z doświadczeń przy pisaniu specyfikacji protokołu HTTP dla systemów rozproszonych. REST wykorzystuje m.in. jednorodny interfejs i bezstanową komunikację.

### **Sesja Linuksowa** (również Sesja / konferencja / SL)

Jest to konferencja poświęcona rozwiązaniom Open Source oraz najnowszym trendom w systemach z rodziny GNU/Linux. Sesja Linuksowa jest corocznym wydarzeniem organizowanym przez ASI, na którym, w przeciągu dwóch dni kilkunastu prelegentów wygłasza wykłady związane z wolnym oprogramowaniem.

### **Cel**

Celem projektu jest stworzenie mobilnej alternatywy dla strony internetowej Sesji Linuksowej. Aplikacja ma przedstawiać tematykę i czas trwania poszczególnych wykładów oraz dodatkowe informacje związane z konferencją.

## Zakres

Aplikacja ma na celu przedstawianie następujących informacji w stanie jak najbardziej aktualnym:

- Prelekcje (wykłady)
  - Tytuł
  - Opis
  - Czas trwania
  - Prelegent
- Prelegenci
  - Imię i nazwisko, [pseudonim]
  - Krótki opis (notka biograficzna)
  - Zdjęcie
  - Prowadzona prelekcja
- Partnerzy
  - Nazwa organizacji / firmy / spółki
  - Typ (Sponsor / Partner / Organizator)
  - Logo
  - Odnośnik do strony internetowej
- Informacje o konferencji
  - Podstawowe informacje, założenia, cel Sesji
  - Kontakt do organizatorów i twórcy aplikacji
  - Informacje dotyczące Middle Party – nieformalnej imprezy networkingowej towarzyszącej konferencji

# **Wymagania**

Aplikacja musi:

1. Zawsze wyświetlać aktualne dane dotyczące konferencji (mogą się zmieniać w nieokreślonych momentach, nawet w czasie trwania prelekcji)
2. Działać płynnie i wyglądać estetycznie i w zgodzie z Material Design na wszystkich nowoczesnych urządzeniach Android (wsparcie dla API $\geq 21$ ). <sup>App.[1]</sup>
3. Nie wymagać połączenia z internetem przez większość czasu użytkowania aplikacji.
4. Być zgodna z projektem graficznym (kolorystyka, styl) strony internetowej SL.
5. Być w pełni zgodna z ideą wolnego oprogramowania (całość kodu dostępna publicznie)

Aplikacja nie musi:

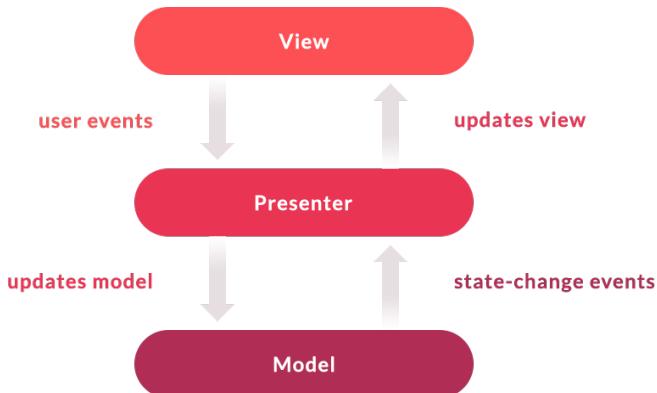
1. Wspierać urządzeń Android API $< 21$ .
2. Wspierać prelekcji prowadzonych przez wielu prelegentów i prelegentów prowadzących wiele prelekcji.
3. Dostarczać anglojęzycznej wersji aplikacji
4. Obsługiwać serwerów SMTP (kontakt poprzez odesłanie użytkownika do aplikacji e-mail)

# **Architektura Systemu**

System powinien składać się z aplikacji mobilnej na system Android, napisanej w Android SDK w języku Java 1.8 (z użyciem kompilatora JACK) oraz z RESTful API napisanego w Flask w języku Python z wykorzystaniem bazy danych SQLAlchemy. API jest wykonywane jako aplikacja webowa na serwerze ASI pod adresem <http://tramwaj.asi.wroclaw.pl> na porcie 6937. Komunikacja przebiega poprzez zastosowanie nesztyfrowanego połączenia http. Ciało odpowiedzi http zapisywane są w formacie JSON. API obsługuje również odpowiedzi błędów o różnych kodach (m.in. 404, 500, 502) w formacie JSON, zamiast domyślnych odpowiedzi http.

# Architektura Aplikacji

Aplikacja na system Android zaprojektowana jest w oparciu o wzorzec projektowy Model View Presenter w celu zapewnienia modularności aplikacji i rozwoju istniejących funkcjonalności oraz dodawania nowych w przyszłości.



Aby zapewnić kompatybilność Widoków <? extends Fragment> z prezenterami <POJO> zastosowane są Interfejsy nazywane Kontraktami (Contract).

Aby zapewnić hermetyzację poszczególnych modułów aplikacji, domyślnym modyfikatorem dostępu dla wszystkich metod komunikujących Widok z Prezenterem jest package private.

Nawigację i komunikację pomiędzy fragmentami nadzoruje Aktywność będąca rodzicem wszystkich podstawowych ekranów w aplikacji i zajmująca się akcjami wymagającymi użycia Intentów.

W celu zwiększenia modularności i przejrzystości kodu użyte są następujące open-source'owe rozwiązania niebędące częścią projektu:

Dagger2 – wstrzykiwanie zależności

ButterKnife – wstrzykiwanie widoków

Retrofit – odwzorowanie RESTful API w modelach Aplikacji, komunikacja http (okHTTP3), parsowanie odpowiedzi JSON na POJO (gson)

SugarORM – operacje bazodanowe na lokalnej bazie danych SQLite

Picasso – wyświetlanie, formatowanie, optymalizacja, cache'owanie, pobieranie zdjęć z internetu.

Aplikacja w momencie włączenia wyświetla SplashScreen – ikonę SL ładującą się zgodnie z postępami w uruchamianiu aplikacji. Aplikacja porównuje czas ostatniej aktualizacji zapisany w SharedPreferences oraz na serwerze i w przypadku niezgodności pobiera zmodyfikowane informacje aktualizując bazę danych – zapewnia to mniejsze wykorzystanie internetu niż każdorazowe pobranie pełnego zestawu danych oraz większą szybkość uruchamiania aplikacji poprzez zmniejszenie objętości zapytań http i częstotliwości aktualizacji bazy danych.

# Funkcjonalności

Aplikacja zawiera następujące funkcjonalności:

- Agenda
  - Wyświetlanie podstawowych informacji na temat prelekcji: tytuł, godziny i dzień trwania, imię i nazwisko prelegenta
  - Możliwość wyświetlenia bardziej szczegółowych informacji na temat prelekcji: opis, zdjęcie prelegenta
  - Możliwość podzielenia się informacją o prelekcji na portalach społecznościowych (wspariane Twitter, LinkedIn, oraz dowolne, wybrane przez użytkownika portale na zasadzie systemowego Intentu)
  - Wyświetlenie w liście z wyraźnym sekcjonowaniem na prelekcje odbywające się w różnych dniach, posortowane według daty i godziny rozpoczęcia prelekcji
- Prelegenci
  - Wyświetlenie podstawowych informacji na temat prelegenta: imię i nazwisko, zdjęcie
  - Możliwość wyświetlenia bardziej szczegółowych informacji na temat prelegenta: opis (krótka notka biograficzna), prowadzona prelekcja (tytuł oraz dzień i godzina)
  - Wyświetlenie w postaci siatki
- Middle Party
  - Czym jest Middle Party, Kiedy odbywa się MP, Jaka jest motywacja MP
  - Gdzie się odbywa Middle Party
    - Możliwość bycia prowadzonym na miejsce odbycia się imprezy z Google Maps
  - Wyświetlenie w postaci kart rozwijających się w pełnoekranowe dialogi
- O Sesji
  - Czym jest Sesja, Jak wygląda Sesja, Jaki jest cel Sesji
  - Formularz kontaktowy
    - Wiadomość e-mail do organizatorów lub twórcy aplikacji (Intent do aplikacji e-mail)
  - Wyświetlenie w postaci kart rozwijających się w dialogi z akcjami
- Partnerzy
  - Wyświetlenie informacji na temat partnera: nazwa, typ partnerstwa, logo
  - Możliwość przejścia na stronę internetową partnera
  - Wyświetlenie w postaci listy kart

# Interfejs użytkownika

Interfejs użytkownika podzielony jest na dwie zasadnicze aktywności: SplashActivity i MainActivity. W MainActivity wyświetlane są Fragmenty poszczególnych modułów. Poniżej przedstawione są różne stany wszystkich ekranów.

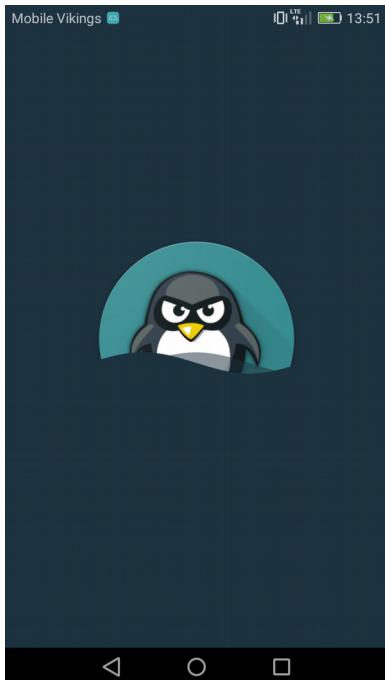


fig. 1 : SplashScreen

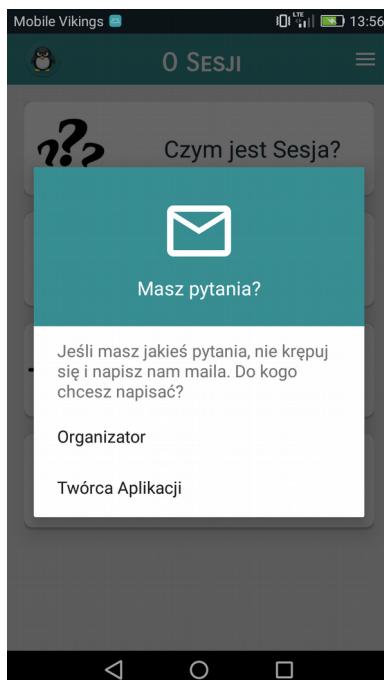


fig. 2 : O Sesji dialog akcje

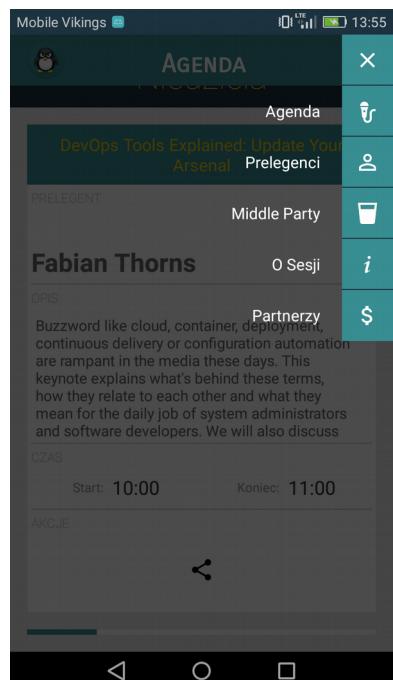


fig. 3 : Menu Główne

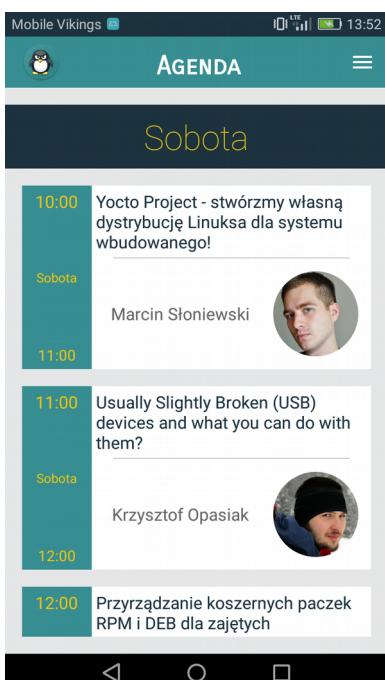


fig. 4 : Agenda lista

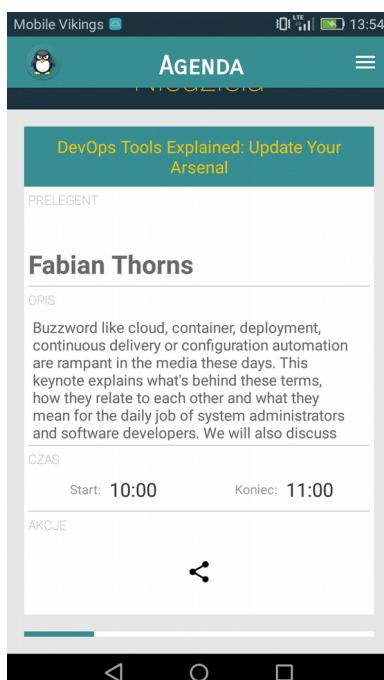


fig. 5 : Agenda szczegółowy

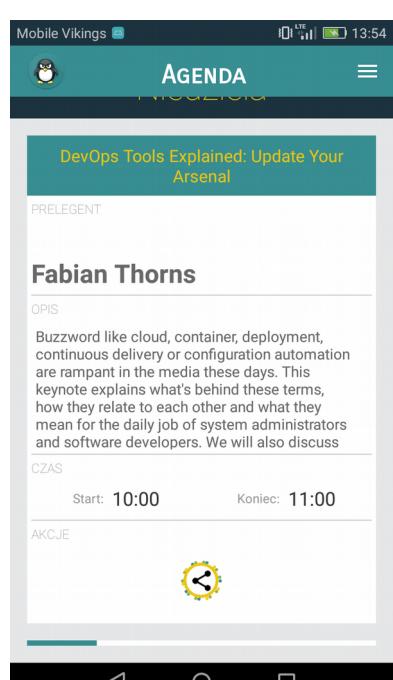


fig. 6 : Agenda share efekt

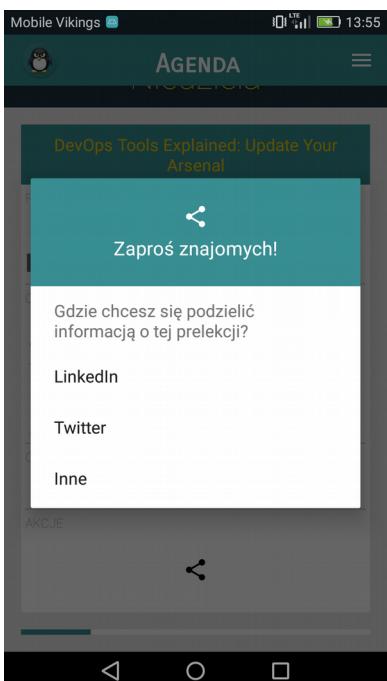


fig. 7 : Agenda share dialog

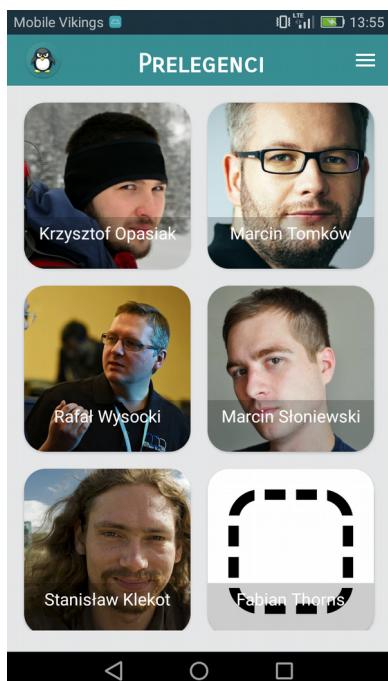


fig. 8 : Prelegenci siatka



fig. 9 : Prelegenci szczegóły



fig. 10 : MiddleParty ekran

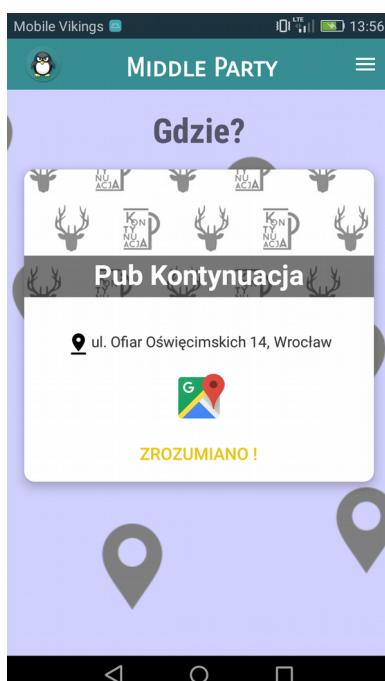


fig. 11 : MiddleParty gdzie



fig. 12 : MP czym jest



fig. 13 : Partnerzy



fig. 14 : O Sesji ekran

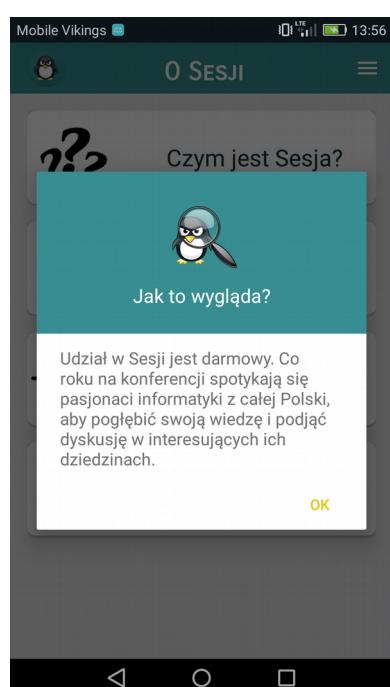


fig. 15 : O Sesji dialog